



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/712,640	11/12/2003	Rameshkumar G. Illikkal	42P16966	8973
8791	7590	06/30/2008	EXAMINER	
BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP			MERED, HABTE	
1279 OAKMEAD PARKWAY			ART UNIT	PAPER NUMBER
SUNNYVALE, CA 94085-4040			2616	
MAIL DATE		DELIVERY MODE		
06/30/2008		PAPER		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/712,640	Applicant(s) ILLIKKAL, RAMESHKUMAR G.
	Examiner HABTE MERED	Art Unit 2616

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
 - If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
 - Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 6/11/2008.
- 2a) This action is FINAL. 2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-10,13-18,20-22,24 and 25 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) Claim(s) _____ is/are allowed.
- 6) Claim(s) 1-10,13-18,20-22 and 24-25 is/are rejected.
- 7) Claim(s) _____ is/are objected to.
- 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on 12 November 2003 is/are: a) accepted or b) objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) Notice of References Cited (PTO-892)
- 2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) Information Disclosure Statement(s) (PTO/SB/08)
 Paper No(s)/Mail Date _____
- 4) Interview Summary (PTO-413)
 Paper No(s)/Mail Date _____
- 5) Notice of Informal Patent Application
- 6) Other: _____

DETAILED ACTION

1. The amendment filed on 6/11/2008 has been entered and fully considered.
2. Claims 1-10, 13-18, 20-22, and 24-25 are pending. Claims 1, 8, 14, and 21 are the base independent claims. Claims 11, 12, 19, and 23 are previously cancelled.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. **Claims 1, 2, 3, 7-9, 14-17, 20-22, and 24-25** are rejected under 35 U.S.C. 103(a) as being unpatentable over Beier et al (US Pub. No.2003/0065812 A1) in view of Brustoloni et al (US 6, 625, 149 B1) and Ganfield (US Pub. No. 20040218631).

Regarding **claim 1**, Beier'812 discloses a method comprising: receiving a packet at a network device (**Figure 4, element 400 is the network device and network packet 305 of Figure 3A shows the packet being received**);

pre-fetching a protocol control block (PCB) (i.e. Beier'812 refers to PCB as “packet descriptor” – see paragraph 36 and element 705 in Figure 7) associated with the received packet into a cache of a selected processing unit (In Figure 3A, at time t2 packet is cached in entry 323 in unified cache – cache is shared by all processors to enhance memory access**);**

and retrieving the PCB from the cache of the selected processing unit when the selected processing unit is ready to process the packet (**Figure 3B processor 330**

accesses PCB or packet descriptor from cache 320. It should be noted that Beier'812 teaches receiving a packet and before processing it a check is made if a PCB associated with the packet exists and if none exists the PCB is pre-fetched as indicated blocks 680 and 690 of Figure 6. Also see t3 in Figure 7 and paragraphs 65 and 70).

Beier'812 fails to disclose queuing the received packet for processing.

However, the above mentioned claimed limitations are well known in the art as evidenced by Brustoloni'149. In particular, Brustoloni'149 discloses queuing the received packets for processing. (**See Figure 3, element 80 queuing received network packets)**

In view of the above, having the method of Beier'812 and then given the well established teaching of Brustoloni'149, it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the method of Beier'812 as taught by Brustoloni'149, the motivation for queuing the received packets being to prevent the overflow of the input buffers.

Beier'812 also fails to disclose a method further comprising pre-fetching a header associated with the packet into the cache of the selected processing unit.

However, the above mentioned claimed limitations are well known in the art as evidenced by Garnfield'631. In particular, Garnfield'631 discloses queuing a method, further comprising pre-fetching a header associated with the packet in the cache of the selected processing unit. (**Figure 5B, elements 530 and 526 and Paragraphs 32 and**

38 show header of received packets being placed in cache as part of the packet descriptor)

In view of the above, having the method of Beier'812 and then given the well established teaching of Garnfield'631, it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the method of Beier'812 as taught by Garnfield'631, the motivation to save the header associated with the packet into the cache memory is to minimize processor time by minimizing latency caused by unnecessary repeated access to memory as implied by Garnfield'631 in Paragraphs 38 and 39.

Regarding **claim 2**, Beier'812 discloses a method wherein the PCB is pre-fetched in response to receiving the received packet before processing the received packet (**Beier'812 teaches receiving a packet and before processing it a check is made if a PCB associated with the packet exists and if none exists the PCB is pre-fetched as indicated blocks 680 and 690 of Figure 6. Also see t3 in Figure 7 and paragraphs 65 and 70).**

Regarding **claim 3**, the combination of Beier'812, Garnfield'631, and Brustoloni'149 teaches a method of further comprising retrieving the packet header from the cache associated with the selected processor unit when the processing unit is ready to process the packet (**Garnfield'631 shows in paragraphs 39 and 50 the header is retrieved for executing the packet from wherever it is queued**).

Regarding **claim 7**, Beier'812 discloses a method, further comprising processing the packet. (**In Figures 7-9 Beier'812 shows the packet being processed for transmission or forwarding**)

Regarding **claim 8**, Beier'812 discloses an apparatus (**See Figure 4, element 400**) comprising: a receive unit to receive a packet (**Figure 4, element 420**); a pre-fetch unit (**Figure 4, elements 460**) coupled to the receive unit (**Figure 4, element 420**) to pre-fetch a protocol control block (PCB) associated with the packet into a cache (**Figure 3, element 320**) of a processing unit (**any of the processors/modules in Figure 4 can be the selected processors and each of them can have their own PCB cache as illustrated in paragraphs 6, 48, 53 and 76 abstract but Beier'812 advocates a unified cache to synchronize and minimize latency**); and the processing unit (**Figure 4, element 415**) coupled to the pre-fetch Unit to retrieve the PCB from the cache and process the packet (**Figure 4, elements 460 and Figure 7**).

Beier'812 fails to disclose queuing the received packet for processing.

However, the above mentioned claimed limitations are well known in the art as evidenced by Brustoloni'149. In particular, Brustoloni'149 discloses queuing the received packets for processing. (**See Figure 3, element 80 queuing received network packets**)

In view of the above, having the apparatus of Beier'812 and then given the well established teaching of Brustoloni'149, it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the apparatus of

Beier'812 as taught by Brustoloni'149, the motivation for queuing the received packets being to prevent the overflow of the input buffers.

Beier'812 also fails to disclose an apparatus further comprising pre-fetching a header associated with the packet into the cache of the selected processing unit.

However, the above mentioned claimed limitations are well known in the art as evidenced by Garnfield'631. In particular, Garnfield'631 discloses queuing a method, further comprising pre-fetching a header associated with the packet in the cache of the selected processing unit. (**Figure 5B, elements 530 and 526 and Paragraphs 32 and 38 show header of received packets being placed in cache as part of the packet descriptor**)

In view of the above, having the apparatus of Beier'812 and then given the well established teaching of Garnfield'631, it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the apparatus of Beier'812 as taught by Garnfield'631, the motivation to save the header associated with the packet into the cache memory is to minimize processor time by minimizing latency caused by unnecessary repeated access to memory as implied by Garnfield'631 in Paragraphs 38 and 39.

Regarding **claim 9**, the combination of Beier'812, Ganfield'631, and Brustoloni'149 discloses an apparatus and system where the receive unit is a network interface card. (**Beier'812 Figure 4, element 420 and also Brustoloni'149 in Column 4:25-26**)

Regarding **claim 14**, Beier'812 discloses an article of manufacture comprising: a machine accessible medium including content that when accessed by a machine causes the machine to:

receive a packet (**Figure 3A shows the network packet 305 being received**);
pre-fetch a protocol control block (PCB) (i.e. Beier'812 refers to PCB as “packet descriptor” – see paragraph 36 and element 705 in Figure 7) associated with the packet and packet header (**In Figure 3A, at time t2 packet is cached in entry 323 in unified cache**);

retrieve the PCB from the cache when the processing unit is ready to process the packet (**Figure 3B processor 330 accesses PCB or packet descriptor from cache 320**);

and to pre-fetch a PCB for packet to be sent when the to-be sent packet (**See paragraph 47**) is queued for transmission across a network (**see Figure 9 for actual forwarding 920**) wherein the PCB for the to-be-sent packet is pre-fetched in response to a send request being initiated for the to-be-sent packet (**It should be noted that Beier'812 teaches receiving a packet and before processing it a check is made if a PCB associated with the packet exists and if none exists the PCB is pre-fetched as indicated blocks 680 and 690 of Figure 6. Also see t3 in Figure 7 and paragraphs 65 and 70**).

Beier'812 fails to disclose queuing the received packet for processing.

However, the above mentioned claimed limitations are well known in the art as evidenced by Brustoloni'149. In particular, Brustoloni'149 discloses queuing the

received packets for processing. (**See Figure 3, element 80 queuing received network packets**)

In view of the above, having the machine of Beier'812 and then given the well established teaching of Brustoloni'149, it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the machine of Beier'812 as taught by Brustoloni'149, the motivation for queuing the received packets being to prevent the overflow of the input buffers.

Beier'812 also fails to disclose a machine further comprising pre-fetching a header associated with the packet into the cache of the selected processing unit.

However, the above mentioned claimed limitations are well known in the art as evidenced by Garnfield'631. In particular, Garnfield'631 discloses queuing a method, further comprising pre-fetching a header associated with the packet in the cache of the selected processing unit. (**Figure 5B, elements 530 and 526 and Paragraphs 32 and 38 show header of received packets being placed in cache as part of the packet descriptor**)

In view of the above, having the machine of Beier'812 and then given the well established teaching of Garnfield'631, it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the machine of Beier'812 as taught by Garnfield'631, the motivation to save the header associated with the packet into the cache memory is to minimize processor time by minimizing latency caused by unnecessary repeated access to memory as implied by Garnfield'631 in Paragraphs 38 and 39.

Regarding **claim 15**, the combination of Beier'812, Garnfield'631, and Brustolon'i149 disclose an article of manufacture, wherein the machine-accessible medium further includes content that causes the machine to pre-fetch the header associated with the packet in the cache (**Garnfield'631 Figure 5B, elements 530 and 526 and Paragraphs 32, 38 and 39 show header of received packets being placed in cache as part of the packet descriptor**).

Regarding **claim 16**, the combination of Beier'812, Garnfield'631, and Brustolon'i149 discloses an article of manufacture and system, further comprising retrieving the packet header from the cache when the processing unit is ready to process the packet (**Garnfield'631 shows in paragraphs 39 and 50 the header is retrieved for executing the packet from wherever it is queued**).

Regarding **claim 17**, it is noted that the limitations of claim 17 corresponds to that of claim 7 as discussed above, please see the Examiner's comments with respect to claim 7 as set forth in the rejection above.

Regarding **claim 20**, Beier'812 discloses an article of manufacture, further comprising storing the packet in a memory coupled to the processing unit (**See Figure 4, memory element 430 coupled to all processors and see paragraph 48 for details**).

Regarding **claim 21**, Beier'812 discloses a system comprising: a receive unit to receive a packet (**Figure 4, element 420**); a memory coupled to the receive unit to store the received packet (**Figure 4, elements 430, 440, 450**); a memory controller coupled to the memory to manage the memory (**Figure 4 memory devices have to have some**

form controller); a pre-fetch unit (Figure 4, element 460 and see also Figure 7) coupled to the receive unit to pre-fetch a protocol control block (PCB) associated with the packet into a cache(**Figure 4, element 430 and see paragraph 48**); and a processing unit to retrieve the PCB from the cache and process the packet (**Figure 4, element 460 and see also Figure 7 for more details on accessing cache**).

Beier'812 fails to disclose queuing the received packet for processing.

However, the above mentioned claimed limitations are well known in the art as evidenced by Brustoloni'149. In particular, Brustoloni'149 discloses queuing the received packets for processing. (**See Figure 3, element 80 queuing received network packets**)

In view of the above, having the system of Beier'812 and then given the well established teaching of Brustoloni'149, it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the system of Beier'812 as taught by Brustoloni'149, the motivation for queuing the received packets being to prevent the overflow of the input buffers.

Beier'812 also fails to disclose a system further comprising pre-fetching a header associated with the packet into the cache of the selected processing unit and retrieve the packet header when the packet is ready.

However, the above mentioned claimed limitations are well known in the art as evidenced by Garnfield'631. In particular, Garnfield'631 discloses queuing a method, further comprising pre-fetching a header associated with the packet in the cache of the selected processing unit (**Figure 5B, elements 530 and 526 and Paragraphs 32 and**

38 show header of received packets being placed in cache as part of the packet descriptor) and retrieve the packet header when the packet is ready (**Garnfield'631 shows in paragraphs 39 and 50 the header is retrieved for executing the packet from wherever it is queued**).

In view of the above, having the system of Beier'812 and then given the well established teaching of Garnfield'631, it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the system of Beier'812 as taught by Garnfield'631, the motivation to save the header associated with the packet into the cache memory is to minimize processor time by minimizing latency caused by unnecessary repeated access to memory as implied by Garnfield'631 in Paragraphs 38 and 39.

Regarding **claim 22**, it is noted that the limitations of claim 22 corresponds to that of claim 9 as discussed above, please see the Examiner's comments with respect to claim 9 as set forth in the rejection above.

Regarding **claim 24**, the combination of Beier'812, Garnfield'631, and Brustoloni'149 discloses a system further comprising a send unit (**i.e. Figure 9, element 920**) to pre-fetch a PCB (**Figure 9, see time t9**) for a packet to be sent when the to-be-sent packet is queued (**See paragraph 42 indicating inter-process queue**) for transmission across the network (**See Figure 9, time t12 to be queued at a tx filter for transmission. Of course, Garnfield'631 teaches queuing received packets for transmission**).

Regarding **claim 25**, the combination of Beier'812, Garnfield'631, and Brustoloni'149 discloses a system wherein the PCB for the to-be-sent packet is pre-fetched in response to a send request being initiated for the to-be-sent packet (**Figure 9 shows in Paragraph 73 that the network device receiving a packet to be routed initiates the route look-up and transmission constituting the send-request process**).

5. **Claims 4-6, 10, 13, and 18** are rejected under 35 U.S.C. 103(a) as being unpatentable over Beier'812 in view of Garnfield'631 and Brustoloni'149 as applied to claims 1, 8, 14, and 21 above, and further in view of Kaniyar et al (US 7, 219, 121 B2).

Regarding **claim 4**, the combination of Beier'812, Garnfield'631, and Brustoloni'149 discloses a method further comprising sending an interrupt to notify the processing unit of the receipt of the packet (**See Brustoloni'149 Column 5, Lines 33-37 and Figure 3, element 40 (i.e. interrupt unit)**).

However, the combination of Beier'812, Garnfield'631, and Brustoloni'149 fail to expressly disclose a method further comprising sending an interrupt to notify the selected processing unit of the receipt of the packet.

However, the above mentioned claimed limitations are well known in the art as evidenced by Kaniyar'121. In particular, Kaniyar'121 discloses a method further comprising sending an interrupt to notify the selected processing unit of the receipt of the packet. (**See Figure 4, steps 408, 410, 412 and the abstract and Column 1,**

**Lines 44-67 and Column 2, Lines 34-67 and Column 3, Lines 1-30 and Column 6,
Lines 46-67 and Column 8, Lines 15-67)**

In view of the above, having the method based on the combination of Beier'812, Garnfield'631, and Brustoloni'149 and then given the well established teaching of Kaniyar'121, it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the method based on the combination of Beier'812, Garnfield'631, and Brustoloni'149 as taught by Kaniyar'121, the motivation being to select the destination of an interrupt is to ensure the same processor is always selected in order to ensure that Input/Output tasks associated with a particular connection are processed by the same processor as indicated by Kaniyar'121 in Column 2:40-45.

Regarding **claim 5**, the combination of Beier'812, Garnfield'631, Brustoloni'149 and Kaniyar'121 teach a method wherein the interrupt is software. (**See Kaniyar'121 Column 5, Lines 63-67 and Column 6, Lines 46-52 and Kaniyar'121 Deferred Procedure Call (DPC) is a software interrupt as a procedure call is simply a function call in Microsoft Operating software.**)

Regarding **claim 6**, the combination of Beier'812, Garnfield'631, Brustoloni'149 and Kaniyar'121 discloses a method, further comprising storing the packet in a memory coupled to the processing unit (**See Figure 4, memory element 430 coupled to all processors and see paragraph 48 for details**).

Regarding **claim 10**, the combination of Beier'812, Garnfield'631, and Brustoloni'149 fails to disclose an apparatus further comprising an interrupt service unit

to check the destination of the interrupt, disable further interrupts from the network interface card, initiate a software interrupt, and queue the packet for processing.

However, the above mentioned claimed limitations are well known in the art as evidenced by Kaniyar'121. In particular, Kaniyar'121 discloses an apparatus further comprising an interrupt service unit (**Figure 3, element 327 and Column 5:64-67 and Column 6:44-67**) to check the destination of the interrupt, disable further interrupts from the network interface card, initiate a software interrupt, and queue the packet for processing. (**See Figure 4, steps 408, 410, 412 and the abstract and Column 1, Lines 44-67 and Column 2, Lines 34-67 and Column 3, Lines 1-30 and Column 6, lines 46-67 and Column 8, lines 15-67)**)

In view of the above, having the apparatus based on the combination of Beier'812, Garnfield'631, and Brustoloni'149 and then given the well established teaching of Kaniyar'121, it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the apparatus based on the combination of Beier'812, Garnfield'631, and Brustoloni'149 as taught by Kaniyar'121, the motivation being to select the destination of an interrupt is to ensure the same processor is always selected in order to ensure that Input/Output tasks associated with a particular connection are processed by the same processor as indicated by Kaniyar'121 in Column 2:40-45.

Regarding **claim 13**, it is noted that the limitations of claim 13 corresponds to that of claim 4 as discussed above, please see the Examiner's comments with respect to claim 4 as set forth in the rejection above.

Regarding **claim 18**, it is noted that the limitations of claim 18 corresponds to that of claim 4 as discussed above, please see the Examiner's comments with respect to claim 4 as set forth in the rejection above.

Response to Arguments

6. Applicant's arguments with respect to claim 1 have been fully considered but are moot in view of the new ground(s) of rejection and in view of the newly amended claim 1.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to HABTE MERED whose telephone number is (571)272-6046. The examiner can normally be reached on Monday to Friday 9:30AM to 5:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Aung S. Moe can be reached on 571 272 7314. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Aung S. Moe/
Supervisory Patent Examiner, Art Unit 2616

/Habte Mered/
Examiner, Art Unit 2616